Bayesian Q-learning with Assumed Density Filtering

Heejin Jeong and Daniel D. Lee
Department of Electrical and System Engineering, University of Pennsylvania

INTRODUCTION

Bayesian Reinforcement Learning (BRL)

Markov Decision Process \( M = \langle S, A, P, R, \gamma \rangle \)

- Goal: To maximize its expected total discounted future reward
  - Value: \( V^\pi(s) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s] \)
  - Action-Value: \( Q^\pi(s, a) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s, a_0 = a] \)

Bayesian Q-learning with Assumed Density Filtering (ADFAQ)

Q-learning

The most popular off-policy TD learning - After observing a reward \( r_t \) and the next state \( s_{t+1} \),

\[
Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[ r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t) \right]
\]

Belief Updates on Q-values

- \( Q^\pi(s, a) \approx N(\mu_{s,a}, \sigma_{s,a}^2) \) where \( \mu_{s,a} \neq \mu_{s',a'} \), \( \sigma_{s,a}^2 \neq \sigma_{s',a'}^2 \) if \( s \neq s' \) or \( a \neq a' \)
- For one-step Temporal Difference (TD) Learning, we observe \( s', r \)

\[
\begin{align*}
& p_{\text{Prior}}(q|\theta, s', r) \propto \text{Likelihood} \times \text{Prior} \\
& \propto p(q' | y' = r - \gamma Q(s'), \theta) p_{\text{Prior}}(q|\theta)
\end{align*}
\]

For each next possible action \( a \):

Approximated ADFAQ (ADFAQ-Approx)

When \( \sigma^2 \ll 1 \), \( \phi(z) \approx \delta(z) \) (dirac delta function) and \( \Phi(z) \approx H(z) \) (Heaviside function).

Define a function \( f(z) \) - the approximation of the term inside the summation, \( c_{i,j} \phi(\theta) \Phi(z) :\)

\[
f(q|\gamma, \sigma) = \begin{cases} 
\frac{1}{\sqrt{2\pi \sigma^2}} & \text{for } q \in [-\mu_{s,a} + \epsilon, \mu_{s,a} + \epsilon] \\
0 & \text{otherwise}
\end{cases}
\]

Then, \( p_{\text{Prior}}(q|\theta, r, s) \approx p_{\text{Prior}}(q|\theta) = \frac{1}{2} \sum_{i,j} c_{i,j} \phi(\theta) \Phi(z) \) for \( z \in (\infty, -\infty) \)

Applying ADF, new mean and variance are:

\[
\begin{align*}
\mu_{s,a}^{\text{new}}(q|\theta, r, s) &= \mu_{s,a}^{\text{old}}(q|\theta, r, s) + \sum_{i,j} c_{i,j} \phi(\theta) \Phi(z) \\
\sigma_{s,a}^2^{\text{new}}(q|\theta, r, s) &= \sigma_{s,a}^2^{\text{old}}(q|\theta, r, s) + \sum_{i,j} c_{i,j} \phi(\theta) \Phi(z)
\end{align*}
\]

Just a linear combination of IV mean/variance

Algorithm Complexity

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Time per step</th>
<th>Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q-learning</td>
<td>( O(</td>
<td>A</td>
</tr>
<tr>
<td>ADFAQ-Approx</td>
<td>( O(</td>
<td>S</td>
</tr>
<tr>
<td>ADFAQ-App</td>
<td>( O(</td>
<td>S</td>
</tr>
</tbody>
</table>

Connection to Q-learning

Suppose that \( c_{i,j} = 0 \forall i \neq \arg \max_j \mu_{i,j} \) we can correspond the learning rate of Q-learning to the following:

\[
\alpha = \frac{\gamma}{1 + (\gamma |\phi(\theta)\Phi(z)|)^2}
\]